

Configuração Definitiva do Samba

Sumário

1. Vantagens.....	5
2. Desvantagem.....	5
3. Conclusão Final.....	5
4. Vamos botar a mão na massa!	6
5. Instalação.....	6
6. Configuração – Parte 1 (Criando os diretórios).....	6
7. Configuração – Parte 2 (Arquivo de Configuração do Samba)	6
8. Configuração – Parte 3 (Configurando Permissões).....	9
8.1. Configuração – Parte 3 (Configurando Permissões – Usuários e Grupo).....	9
8.2. Configuração – Parte 3 (Configurando Permissões – Permissões de acesso e segurança)	10
8.3. Configuração – Parte 3 (Configurando Permissões – Realizando as alterações no arquivo de configuração do samba, e “botando tudo para funcionar”)	13

Introdução

Para o compartilhamento de diretórios via rede, com a possibilidade de configurar permissões de controle de acesso, o Samba sobressai-se sobre os concorrentes, além de que o mesmo também é capaz de compartilhar diretórios de um sistema de arquivos Linux (ext,jfs...) através da rede, possibilitando o acesso ao mesmo utilizando o protocolo cliente “smb”, para distribuições Linux, e o próprio Windows Explorer (explorador de diretórios), para sistemas operacionais Microsoft Windows.

1. Vantagens

Em relação ao seu concorrente proprietário (\$\$), o Sistema Operacional Microsoft Windows Server, o Samba é muitíssimo mais completo, apresenta menos erros (conhecidos também como bugs), além do que, como o software “samba” deve ser instalado em um servidor Linux, a possibilidade de ter o servidor infectado por malware (vírus, cavalo de tróia, worm, etc.) é MÍNIMA. E por fim, a configuração de permissão de acesso de diretórios é feita INTEIRAMENTE utilizando ferramentas da distribuição Linux, o que torna muitas das técnicas *hacking* ineficazes.

2. Desvantagem

Então, qual o principal motivo de o mercado utilizar principalmente Windows Server para tal serviço, você se pergunta?

Simple: Em sistemas Windows, configurar o sistema para compartilhar diretórios é brincadeira de criança, já em sistemas Linux, o arquivo de configurações do Samba é assustador para quem nasceu no mundo do “mas onde é que clica?”. Porém, ao término da configuração, o sistema durará por muitos anos (se não eternamente), se tudo depender do Sistema Operacional, pois não há agentes “feitos para destruir o sistema”, como em outros sistemas operacionais proprietários.

3. Conclusão Final

No mundo Windows, a configuração é simples e rápida, porém a mesma precisará ser refeita de tempos em tempos, pois todo tipo de malware está preparado para destruir o sistema, além disso, erros de sistema serão frequentes, confundindo o usuário e o setor de TI da empresa.

Utilizando um servidor Linux, por vias normais, a configuração é mais lenta, porém uma vez terminada a configuração do mesmo, a equipe de TI apenas necessitará pensar em “como agregar funções a mais”. Chega de enxaquecas, e gastos desnecessários com manutenção!

4. Vamos botar a mão na massa!

A configuração será feita em um servidor Debian 7 (Wheezy), porém pode ser adaptada facilmente para CentOS, RedHat, e outras distribuições GNU/Linux. Utilizaremos um terminal modo texto.

5. Instalação

Instale o pacote “samba” em seu servidor Debian.

```
#aptitude install samba
```

ou

```
#apt-get install samba
```

Após a instalação, será criado o arquivo de configurações do servidor samba.

Localização do arquivo:

```
/etc/samba/smb.conf
```

6. Configuração – Parte 1 (Criando os diretórios)

Antes de começar a criar os diretórios que posteriormente estarão disponíveis via rede, planeje no papel quantos departamentos (grupos de pessoas/usuários) sua empresa possui.

Crie um diretório no caminho de preferência, em seu sistema de arquivos.

```
#mkdir /samba
```

Em seguida, crie um subdiretório para cada departamento de sua empresa.

```
#mkdir /samba/marketing
```

```
#mkdir /samba/"departamento pessoal" *
```

```
#mkdir /samba/direção
```

* A forma mais simples de se criar um diretório que contém nome espaçado em Linux é digitando-se apenas o nome espaçado entre aspas (“diretório com nome espaçado”).

Depois de criado os subdiretórios, a etapa 1 estará terminada.

7. Configuração – Parte 2 (Arquivo de Configuração do Samba)

O samba, assim como muitos outros servidores Linux, deve ser configurado alterando os parâmetros presentes em um arquivo de configuração, e estes parâmetros alterados serão então futuramente carregados nas variáveis do software servidor, durante sua inicialização.

O arquivo de configuração do samba encontra-se em:

```
/etc/samba/smb.conf
```

É recomendável renomear o arquivo, pois iniciaremos a configuração de nosso servidor a partir do zero!

```
#mv /etc/samba/smb.conf /etc/samba/smb.conf.original
```

Aqui, utilizaremos o editor de textos nano por ser o editor de textos padrão Linux, isto é, o mesmo encontra-se por padrão em qualquer distribuição Linux, ao contrário do vim, vi, entre outros.

```
#nano /etc/samba/smb.conf
```

Em seu terminal, neste momento, você deve estar visualizando um arquivo novo, recém-criado por você, utilizando o editor de textos vim.

O arquivo de configurações samba está dividido em seções, e cada seção é representada da seguinte forma: “[nome da seção 1]” “[nome da seção 2]” ...

Para configurar-se o samba, iniciaremos pela seção “global”.

```
# A seção global contém parâmetros de configurações globais, os quais serão aplicados a todo o
#servidor, e a todo compartilhamento.
[global]
server string = nomedoserver #Nome Dns
netbios name = nomedoserver #Nome NetBios
workgroup = WORKGROUP #grupo de trabalho das máquinas Windows

#Opções para security:
# none – Nada de senhas!
# user – Requer uma senha Unix, mesmo antes mesmo de escolher o compartilhamento ao qual
#pretende acessar.
# share – Requer uma senha Unix, apenas se ao acessar o compartilhamento você não tiver
#permissões para acessá-lo.
security = share

#No arquivo de log, serão armazenadas informações sobre cada conexão realizada ao servidor.
#'%m' é uma variável que corresponde ao nome da máquina que acessar o servidor samba.
log file = /var/log/samba/samba.log #para um log centralizado
#log file = /var/log/samba/%m.log #para um log por máquina conectada
```

Depois de terminada a configuração global, deve-se configurar os compartilhamentos.

```
#Compartilhando
#[nome do compartilhamento]
[Publicidade e Marketing]
comment = Acesso Restrito ao setor de Marketing
path = /samba/marketing
public = yes #Acesso sem senha, público (yes ou no)
writable = yes #Permitir alterações no diretório? (yes ou no)
#valid users = deixe para mais tarde #Mais tarde!
```

Já é possível testar nossas configurações. Vamos, para isso reiniciar o serviço do samba.

```
#!/etc/init.d/samba restart
```

ou

```
#!/etc/init.d/samba stop
```

```
#!/etc/init.d/samba start
```

ou

```
#!/service samba restart
```

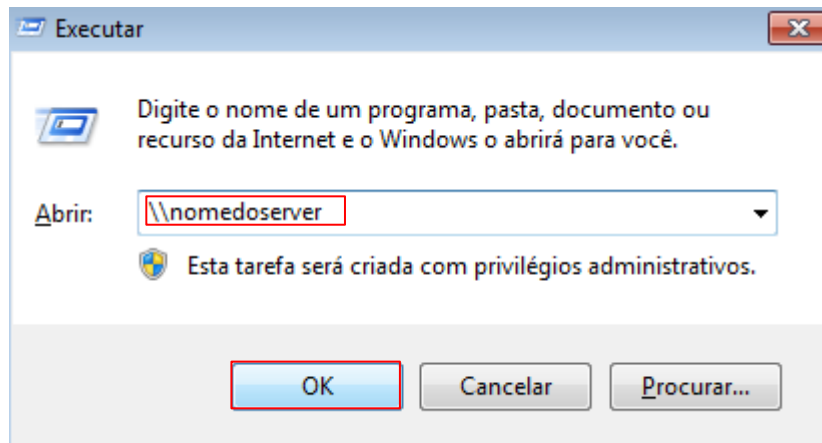
ou

```
#!/service samba stop
```

```
#!/service samba start
```

Vamos testar nossa configuração!

Em um computador Windows, presente na mesma rede e configurado no mesmo grupo de trabalho do samba, chame o diálogo “Executar”, e então digite \\[server string] e aperte “ok”.



Caso tenha escolhido o valor “none ou share” para o parâmetro “security”, nenhum prompt de senha será apresentado.

Caso tenha escolhido o valor “user” para “security”, terminaremos a configuração na seção “Configuração– Parte 3 (Configurando Permissões)”. Você, provavelmente terá acesso com permissões de somente leitura ao diretório “Publicidade e Marketing”. Configuraremos isso mais tarde.

8. Configuração – Parte 3 (Configurando Permissões)

8.1. Configuração – Parte 3 (Configurando Permissões – Usuários e Grupo)

Agora, vamos sair um pouco do arquivo de configurações do samba, e criaremos os usuários que realizarão login via rede através do samba.

Os usuários do samba, assim como citado anteriormente, são usuários comuns Linux.

Vamos criá-los.

```
#useradd joana_dark
#useradd diego_hipolito
#useradd maradonna
```

Os três usuários acima serão usuários do samba, pertencentes ao grupo marketing. Porém, para cada um deles, foi criado um diretório /home. Abaixo seguem os mesmos comandos, porém com parâmetros que impossibilitarão o uso do login e senha para login local no servidor e negarão a criação de uma pasta pessoal (/home/[usuário]).

```
#useradd --no-create-home -s /bin/false joana_dark
#useradd --no-create-home -s /bin/false diego_hipolito
#useradd --no-create-home -s /bin/false maradonna
```

O parâmetro “-s” especifica um shell de comandos ao usuário criado. O shell “/bin/false”, como o nome sugere, é um shell falso, e qualquer usuário que utiliza este shell não será capaz de logar-se corretamente no sistema Linux. Ainda bem que para o usuário acessar os compartilhamentos, ele não precisa de um shell válido, não é mesmo?

Caso você tenha criado um usuário sem especificar o shell, o usuário é configurado com o shell padrão de sua distro. Para alterá-lo, edite diretamente o arquivo de texto “/etc/passwd”, ou execute o comando “usermod -s /bin/false [nomeDoUsuárioExistente]”.

```
#nano /etc/passwd
```

Ou

```
#usermod -s /bin/false [nomeDoUsuárioExistente]
```

Os usuários acima foram criados sem qualquer tipo de senha, pois os usuários do samba não necessitam de uma senha de logon no Linux, portanto configurar uma senha é opcional.

Vamos adicionar estes usuários para serem utilizados no samba.

```
#smbpasswd -a joana_dark
#smbpasswd -a diego_hipolito
#smbpasswd -a maradonna
```

Já se pode testar o login destes usuários em seu servidor samba, utilizando um cliente Microsoft Windows.

Aviso: Logar não quer dizer “acesso garantido aos diretórios”, isso ainda estamos ao passo de configurar.

Para facilitar a administração e ter compartilhamentos samba extremamente seguros, devemos organizar os usuários em grupos. Em uma empresa é muito simples: O nome dos grupos de usuários deverão ser os mesmos de cada departamento da empresa.

Vamos criar o(s) grupo(s).

```
#addgroup marketing
#addgroup departamento_pessoal
#addgroup direcao
```

Neste exemplo, apesar de criarmos 3 grupos, apenas o grupo “marketing” será configurado. Agora, vamos agrupar os funcionários da empresa nos grupos correspondentes, de acordo com o seu departamento. Em nosso caso, os 3 funcionários pertencerão ao mesmo grupo (departamento) “marketing”.

```
#adduser joana_dark marketing
#adduser diego_hipolito marketing
#adduser maradonna marketing
```

8.2. Configuração – Parte 3 (Configurando Permissões – Permissões de acesso e segurança)

Primeiramente, vamos criar um diretório pessoal para cada um dos 3 membros do grupo “marketing”.

```
#mkdir /samba/marketing/maradonna
#mkdir /samba/marketing/diego_hipolito
#mkdir /samba/marketing/joana_dark
```

Aviso: Os nomes dos diretórios não necessitam serem os mesmos que o do usuário.

Primeiramente, ao diretório raiz do departamento iremos configurar quem o “comandar”, e o grupo de usuários que o comandará.

```
#chown root.marketing /samba/marketing
```

O comando “chown” (**change owner**) segue a seguinte sintaxe:
chown [usuário_dono].[grupo_dono] [diretório].

Você deve estar perguntando-se: “Mas por que devemos ter o usuário root de dono, uma vez que o mesmo sempre possui acesso irrestrito à todos os diretórios?”.

E a resposta é simples: O usuário dono, neste momento não será importante. O mais importante é que o grupo marketing agora é dono de seu próprio diretório. Ótimo!

Agora, vamos a uma das partes mais divertidas: A configuração das permissões.

A partir deste momento, o diretório “/samba/marketing” pertence ao usuário “root” e também ao grupo “marketing”, e à todos os membros deste grupo.

É extremamente recomendável conhecer permissões (chmod) à partir deste ponto.

```
#chmod 000 /samba/marketing
```

Sintaxe do comando:

```
chmod [permissão_usuario_dono][permissão_grupo_dono][permissão_para_qualquer_outro_usuario] [diretorio]
```

Para cada permissão (dono, grupo, ou outros usuários), pode-se atribuir um número entre 0 e 7, para permitir o nível de acesso ao diretório especificado.

Abaixo, será explicado cada nível de acesso, sem maiores detalhes.

Permissão	Binário	Decimal
---	000	0
--x	001	1
-w-	010	2
-wx	011	3
r--	100	4
r-x	101	5
rw-	110	6
rwX	111	7

Fonte: <http://www.infowester.com/linuxpermissoes.php>

R – Read (Leitura) – Permissão de “olhar” o conteúdo.

W – Write (Escrita) – Permissão de gravar novos arquivos e alterar os já existentes.

X – Execution (Execução) – Permissão de abrir o diretório e abrir arquivos que estão no mesmo.

Se não compreendeu permissões Linux, você deve estudar sobre a mesma, e o link acima o auxiliará muito em seu aprendizado.

Vamos analisar novamente o comando digitado anteriormente.

```
#chmod 000 /samba/marketing
```

Usuário	Grupo	Outros usuários (não donos)
0	0	0
Leitura=negada, Escrita=negada, Execução=negada.	Leitura=negada, Escrita=negada, Execução=negada.	Leitura=negada, Escrita=negada, Execução=negada.

Sendo assim, ninguém terá acesso ao diretório, correto?

Vamos ser bonzinhos, e liberar o acesso ao usuário e ao grupo? Planejando...

O usuário (root) poderá receber qualquer permissão.

O grupo receberá acesso de leitura e execução (r-x ◇ 5).

Qualquer outro usuário terá seu acesso completamente negado!

```
#chmod 750 /samba/marketing
```

Usuário	Grupo	Outros usuários (não donos)
7	5	0
Leitura=concedida, Escrita=concedida, Execução=concedida.	Leitura=concedida, Escrita=negada, Execução=concedida.	Leitura=negada, Escrita=negada, Execução=negada.

Reinicie os daemons do samba.

```
#/etc/init.d/samba restart
```

Pronto, agora experimente acessar o servidor samba de um cliente Windows, acesse o diretório compartilhado “Marketing”, e uma senha lhe será pedida. Você deve apenas poder ver o conteúdo do diretório, mas não poderá alterá-lo.

Ainda não pode acessar o diretório? Isso é comum, e será resolvido assim que alterarmos algumas configurações do arquivo de configurações do samba, ok? Não se desespere.

Iremos agora tornar os usuários pertencentes ao grupo “marketing” donos de seus próprios subdiretórios.

```
#chown maradonna.marketing/samba/marketing/maradonna
#chown diego_hipolito.marketing/samba/marketing/diego_hipolito
#chown joana_dark.marketing/samba/marketing/joana_dark
```

Agora, cada usuário é dono de um diretório diferente, e, além disso, o grupo também é dono de cada diretório dos usuários do grupo marketing.

Vamos configurar o nível de acesso de cada usuário para cada um dos 3 diretórios.

```
#chmod 750 /samba/marketing/maradonna
#chmod 750 /samba/marketing/diego_hipolito
#chmod 750 /samba/marketing/joana_dark
```

Vamos analisar o comando.

```
#chmod 750 /samba/marketing/joana_dark
```

O usuário dono (joana_dark) recebeu a permissão 7 (acesso total), os membros do grupo “marketing” receberam a permissão 5 (r-x ◊ Read & Execution – Leitura + Execução), permitindo à estes acessarem o diretório livremente, porém não poderão alterar ou salvar quaisquer dados.

8.3. Configuração – Parte 3 (Configurando Permissões – Realizando as alterações no arquivo de configuração do samba, e “botando tudo para funcionar”)

```
#nano /etc/samba/smb.conf
```

Oba! Estamos novamente configurando o smb.conf.
Faremos agora os ajustes finais.

```
#Compartilhando
#[nome do compartilhamento]
[Publicidade e Marketing]
comment = Acesso Restrito ao setor de Marketing
path = /samba/marketing
public = no      #Acesso sem senha, público (yes ou no)
writable = yes   #Permitir alterações no diretório? (yes ou no)
valid users = @marketing #Apenas os membros do grupo marketing acessarão o compartilhamento.
force group = marketing #Força o acesso do grupo marketing somente.
```

Salve o documento, e em seguida, reinicie os daemons do samba.

```
#!/etc/init.d/samba restart
```

Há ainda um parâmetro chamado “veto files”, para você acrescentar à um compartilhamento. O mesmo é útil para vetar nomes de arquivos e/ou extensões, facilitando para o administrador impedir a disseminação de arquivos perigosos que sejam alojados no servidor, e desencorajar os usuários à guardarem arquivos pessoais nos diretórios compartilhados. Veja abaixo um exemplo de uso do parâmetro:

```
veto files = *.exe/*.com/*.scr/*.rar/*.zip/*.ace*.cab/*.bat/*.inf/
```

O parâmetro “veto files” pode ser aplicado à um compartilhamento, ou diretamente na seção global.

Quer mais? Que tal gerenciar o seu servidor com o web admin para o samba, o “swat”?

```
#aptitude install swat
```

Após a instalação, abra seu navegador de internet preferido, e digite o seguinte endereço na barra de endereços de seu navegador: <http://localhost:901>

